

IN THE CLAIMS

1. (Currently Amended) A program office management system, comprising:
~~a program office database storing:~~
computer software stored on a computer readable storage medium and operable to
store informational data associated with accounts, projects, and programs;
store financial data associated with the accounts, projects, and programs;
store schedule and progress data associated with the accounts, projects, and
programs;
store data associated with personnel, roles, and security access information
thereof;
store a plurality of~~tactic table operable to store at least one~~ predefined tactics
~~supported by the program office database and a valid tactic type for each tactic, wherein each of~~
the plurality of the predefined tactics comprises an approach taken to affect change on a
project~~prescribes a change to a project;~~
associate one or more predetermined project milestone categories with at least
some of the plurality of predefined tactics;
~~a tactic type to progress milestone category cross-reference table operable to map~~
~~at least one progress milestone category to the at least one tactic type;~~
store update data associated with the progress, actual expenditures, and labor
resources of the projects and programs;
wherein the data associated with the security access information of personnel
comprise a role definition of a coordinator having authorization to assign one or more persons
to the at least one business unit, assign at least one role to each person, and add projects and
accounts for the at least one business unit;
wherein the data associated with the security access information of personnel
comprise a role table operable to store at least one valid role, and an authorization hierarchical
organization of the at least one valid role, wherein the authorization hierarchical organization is
associated with increasing levels of data access; and
wherein the data associated with the security access information of personnel
associates at least one of the valid roles relevant to the project to each person;
~~at least one user interface operable to~~

display data stored in the program office according to a predetermined security scheme based on the security access information stored in the program office database,

upon selection of a first tactic, comprising one of the plurality of predefined tactics, by a user for use on a particular project, automatically associating with the particular project at least one milestone having a particular milestone category that was previously associated with the first tactic, and further operable to

receive the update data on a periodic basis.

2. (Original) The system, as set forth in Claim 1, wherein the program office database comprises a plurality of relational data structures.

3. (Original) The system, as set forth in Claim 1, wherein the at least one user interface comprises at least one web-based user interface.

4. (Original) The system, as set forth in Claim 1, wherein the at least one user interface comprises at least one self-extracting executable user interface.

5. (Original) The system, as set forth in Claim 1, wherein the at least one user interface comprises at least one program office interface.

6. (Original) The system, as set forth in Claim 1, wherein the program office database comprises more than one copy of the data residing in more than one distributed databases.

7. (Original) The system, as set forth in Claim 1, wherein the user interface comprises more than one copy of the user interface residing in more than one distributed computing system.


8. (Original) The system, as set forth in Claim 1, wherein the data associated with security access information of personnel comprise an assignment table associating a person to at least one role defined within a business unit.

9. (Original) The system, as set forth in Claim 1, wherein the data associated with security access information of personnel comprise an assignment table associating a person to at least one role defined within a business unit, and further to at least one predefined update authority level set by a person having a senior management role within the business unit.

10. (Cancelled)

11. (Cancelled)

12. (Cancelled)



13. (Original) The system, as set forth in Claim 1, wherein the data associated with security access information of personnel comprise a role definition of an account manager capable of having authorization to update account data and project data.

14. (Original) The system, as set forth in Claim 1, wherein the data associated with security access information of personnel comprise a role definition of a project manager capable of having authorization to update project data.

15. (Cancelled)

16. (Original) The system, as set forth in Claim 1, wherein the data associated with translating progress milestones comprise a data table operable to map milestones predefined in a project to milestone categories predefined within the program office database.

17. (Cancelled)

18. (Original) The system, as set forth in Claim 1, wherein the financial data comprise:

a project forecast table operable to store at least one current budget forecast amount for the project; and

a project forecast history table operable to store an original budget forecast amount if it is different than the at least one current budget forecast amount.

19. (Original) The system, as set forth in Claim 1, wherein the financial data comprise:

an account forecast table operable to store at least one revenue and expense budget amount associated with an account; and

an account actual table operable to store at least one revenue and expense actual amount associated with the account.

20. (Original) The system, as set forth in Claim 1, wherein the informational data comprise a project table operable to store informational data associated with at least one project identified by a project identifier.

21. (Original) The system, as set forth in Claim 20, wherein the project table comprises:

a project identifier uniquely identifying each project;

a business unit identifier of a business unit to which the project belongs to;

at least one person identifier of a person assigned a role having a predetermined responsibility for the project; and

a status flag indicative of whether the project is active, pending, or inactive.

22. (Original) The system, as set forth in Claim 1, wherein the information data include an account table comprising:

an account identifier uniquely identifying each account;

a business unit identifier of a business unit to which the account belongs to; and

a person identifier of a person assigned the role of an account manager for the account.

23. (Original) The system, as set forth in Claim 1, wherein the schedule and progress data comprise a milestone actual table operable to store an amount of progress into a specific milestone for a given period for a project.

24. (Original) The system, as set forth in Claim 1, wherein the schedule and progress data comprise:

a project identifier of a project;

a milestone defined for the project;

a reporting period; and

a percentage completion value of the milestone in the reporting period independent of forecast or actuals.

25. (Original) The system, as set forth in Claim 1, wherein the update data comprise:

a project actual table operable to store actual expenditure amounts spent during a specific reporting period for a project; and

a milestone actual table operable to store a percentage completion value of a specific milestone defined for a project during the specific reporting period.

26. (Previously Presented) The system, as set forth in Claim 24, wherein the update data further comprise an account actual table operable to store actual expenditure amounts spent during the specific reporting period for an account.

27. (Previously Presented) The system, as set forth in Claim 1, wherein the program office database further comprises a user weight table operable to store a weight value indicative of importance for each system affected by the projects and programs.

28. (Previously Presented) The system, as set forth in Claim 1, wherein the program office database further comprises a project roadblock table operable to store information about a problem encountered in a project identified by a project identifier and to enable escalated reporting to upper management about unresolved problems.

29. (Previously Presented) The system, as set forth in Claim 28, wherein the project roadblock table comprises:

roadblock type;

date and time that the problem was encountered; and

data on how and when the problem was resolved.

30. (Previously Presented) The system, as set forth in Claim 1, wherein the program office database further comprises a transaction log table operable to record what changes were made to data stored in the program office database, who made the changes, and when the changes were made.

31. (Previously Presented) The system, as set forth in Claim 1, wherein the program office database comprises required data, audit data, program objective specific data, and optional data.

A handwritten signature or set of initials, possibly 'G' or 'H', written in black ink.

32. (Currently Amended) A method of managing a program office, comprising:
storing and accessing data associated with at least one project in a program office database, including informational data, financial data, schedule and progress data associated with the at least one project;

storing update data associated with the at least one project;

identifying persons associated with the at least one project, defining a role hierarchy having roles associated with increasing levels of data access, assigning at least one role relevant to the at least one project to each person, and storing data associated with the persons and their assigned roles in the program office database;

wherein assigning at least one role comprises assigning a role of coordinator, a role having authority to add people for a respective business unit, assign some roles to people, and add projects and accounts of a business unit;

storing a plurality of predefined tactics, wherein each of the plurality of predefined tactics comprises an approach taken to affect change on a project;

associating one or more predetermined project milestone categories with at least some of the plurality of predefined tactics; and

upon selection of a first tactic, comprising one of the plurality of predefined tactics, by a user for use on a particular project, automatically associating with the particular project at least one milestone having a particular milestone category that was previously associated with the first tactic.

~~storing and accessing a tactic table having at least one predefined tactic supported by the program office database, wherein the predefined tactic prescribes a change to a project;~~

~~storing and accessing a tactic type table having at least one valid tactic type;~~


~~storing and accessing a milestone category table having at least one category of milestones; and~~

~~storing and accessing a tactic type to milestone category cross reference table associating the at least one milestone category to the at least one tactic type.~~

33. (Previously Presented) The method, as set forth in Claim 32, wherein identifying persons further comprises assigning an update authorization level to each person by a person having a senior management role.

34. (Previously Presented) The method, as set forth in Claim 33, further comprising restricting and permitting viewing, changing and adding data in the program office database according to the assigned role to each person, rules defined in the program office database, and update authorization level assigned to each person.

35. (Previously Presented) The method, as set forth in Claim 32, wherein assigning at least one role comprises assigning at least one role from the role hierarchy to each person, the roles having increasing capability to access and modify program office database data.



36. (Cancelled)

37. (Cancelled)

38. (Previously Presented) The method, as set forth in Claim 32, wherein assigning at least one role comprises assigning a role of account manager, a role capable of having authority to update project and account data for a respective account.

39. (Previously Presented) The method, as set forth in Claim 32, wherein assigning at least one role comprises assigning a role of project manager, a role capable having authority to update project data for a respective project.

40. (Previously Presented) The method, as set forth in Claim 32, wherein storing and accessing data comprise storing and accessing data stored in at least one relational database.

41. (Previously Presented) The method, as set forth in Claim 32, wherein storing and accessing data associated with the persons and their assigned roles comprise:

storing and accessing an assignment table associating a person identifier to at least one role defined within a specific business unit; and

granting at least one predefined update authority to the person identifier by a person having a predetermined upper management role.

42. (Previously Presented) The method, as set forth in Claim 32, wherein storing and accessing data associated with the persons and their assigned roles comprise storing and accessing a role table having at least one valid role and an authorization hierarchical organization of the at least one valid role.

43. (Cancelled)

44. (Previously Presented) The method, as set forth in Claim 32 further comprising storing and accessing a data table associating a milestone to the at least one tactic.

45. (Previously Presented) The method, as set forth in Claim 32, wherein storing and accessing the financial data comprise:

storing and accessing a project forecast table having at least one current budget forecast amount for the project; and

storing and accessing a project forecast history table operable to store an initial budget forecast amount if it is different than the at least one current budget forecast amount.

46. (Previously Presented) The method, as set forth in Claim 32, wherein storing and accessing the financial data comprise:

storing and accessing an account forecast table operable to store at least one revenue and expense budget amount associated with an account; and

storing and accessing an account actual table operable to store at least one revenue and expense actual amount associated with the account.

47. (Previously Presented) The method, as set forth in Claim 32, wherein storing and accessing the informational data comprise:

storing and accessing a project table operable to store informational data associated with at least one project identified by a project identifier; and

storing and accessing an account table operable to store informational data associated with at least one account identified by an account identifier.

48. (Previously Presented) The method, as set forth in Claim 32, wherein storing and accessing the project table comprise:

storing a project identifier uniquely identifying each project and using the project identifier as a primary key to the project table;

storing and accessing a business unit identifier of a business unit to which the project belongs to;

storing and accessing a person identifier of a person assigned at least one role for the project; and

storing and accessing a status flag indicative of whether the project is active, pending, or inactive.

49. (Previously Presented) The method, as set forth in Claim 32, wherein storing and accessing the account table comprise:

storing and accessing an account identifier uniquely identifying each account;

storing and accessing a business unit identifier of a business unit to which the account belongs to; and

storing and accessing a person identifier of a person assigned the role of an account manager for the account.

50. (Previously Presented) The method, as set forth in Claim 32, wherein storing and accessing the schedule and progress data comprise storing and accessing a milestone actual table having an amount of progress into a specific milestone for a given period for a project.

51. (Previously Presented) The method, as set forth in Claim 32, wherein storing and accessing the schedule and progress data comprise:

storing and accessing a project identifier of a project;

storing and accessing a milestone defined for the project;

storing and accessing a reporting period; and

storing and accessing a percentage completion value of the milestone in the reporting period.

52. (Previously Presented) The method, as set forth in Claim 32, wherein storing and accessing the update data comprise:

storing and accessing a project actual table having actual expenditure amounts spent during a specific reporting period for a project; and

storing and accessing a milestone actual table having a percentage completion value of a specific milestone defined for a project during the specific reporting period.

53. (Previously Presented) The method, as set forth in Claim 52, wherein storing and accessing the update data further comprise storing and accessing an account actual table having actual expenditure amounts spent during the specific reporting period for an account.

54. (Previously Presented) The method, as set forth in Claim 32, further comprising storing and accessing a user weight table having a weight value indicative of importance for each system affected by the projects and programs.

55. (Previously Presented) The method, as set forth in Claim 32, further comprising: storing and accessing a project roadblock table having information about a problem encountered in a project identified by a project identifier; and reporting any problem to management unresolved after a predetermined time period.

56. (Previously Presented) The method, as set forth in Claim 55, wherein storing and accessing the project roadblock table comprise:

storing and accessing a roadblock type;

storing and accessing a date and time that the problem was encountered; and

storing and accessing data on how and when the problem was resolved.

57. (Previously Presented) The method, as set forth in Claim 32, further comprising storing and accessing a transaction log table having what changes were made to data stored in the program office database, who made the changes, and when the changes were made.

58. (Previously Presented) The method, as set forth in Claim 33, wherein storing and accessing the data comprise storing and accessing data via a web browser-based user interface implementing a security scheme using the role and update authorization level assignment to the users.

59. (Previously Presented) The method, as set forth in Claim 33, wherein storing and accessing update data comprise storing the update data via a self-extracting spread sheet-based user interface implementing a security scheme using the role and update authorization level assignment to the users.

60. (Previously Presented) The method, as set forth in Claim 32, further comprising: retrieving data from at least one other data source; and verifying data in the program office database with the data from the at least one other data source.

61. (Previously Presented) The method, as set forth in Claim 32, further comprising: retrieving data from at least one project management tool; and using the data from the at least one project management tool in views, reports, and audits.

62. (Previously Presented) The method, as set forth in Claim 32, further comprising: retrieving data from at least one project management tool; and storing the data from the at least one project management tool in the program office database.

63. (Currently Amended) A system for managing at least one program including a plurality of projects, comprising:

~~at least one program office database storing:~~

computer software stored on a computer readable storage medium and operable to

store informational data associated with projects and programs;

store financial data associated with the projects, and programs;

store schedule and progress data associated with the projects, and programs;

store personnel data associated with persons having responsibility associated with the projects and programs, the personnel data including a unique person identifier for each person;

store security data having an assignment of at least one role to each person and an assignment of at least one update authorization to certain persons having oversight responsibility;

store a plurality of predefined tactics wherein each of the plurality of predefined tactics comprises an approach taken to affect change on a project;

associate one or more predetermined project milestone categories with at least some of the plurality of predefined tactics;

~~a tactic table operable to store at least one predefined tactic supported by the program office database and a tactic type for each tactic, wherein the predefined tactic prescribes a change to a project;~~


~~a tactic type to progress milestone category cross-reference table operable to map at least one progress milestone category to the at least one tactic type; and~~

store update data associated with the progress, actual expenditures, and labor resources of the projects and programs;

wherein the data associated with the security access information of personnel comprise a role definition of a coordinator having authorization to assign one or more persons to the at least one business unit, assign at least one role to each person, and add projects and accounts for the at least one business unit;

~~at least one user interface operable to display and allow access to the data stored in the program office according to a predetermined security scheme based on the person identifier, role and update authorization assignment stored in the at least one program office database;~~

~~and further operable to~~



upon selection of a first tactic, comprising one of the plurality of predefined tactics, by a user for use on a particular project, automatically associating with the particular project at least one milestone having a particular milestone category that was previously associated with the first tactic,

receive the update data on a periodic basis.
